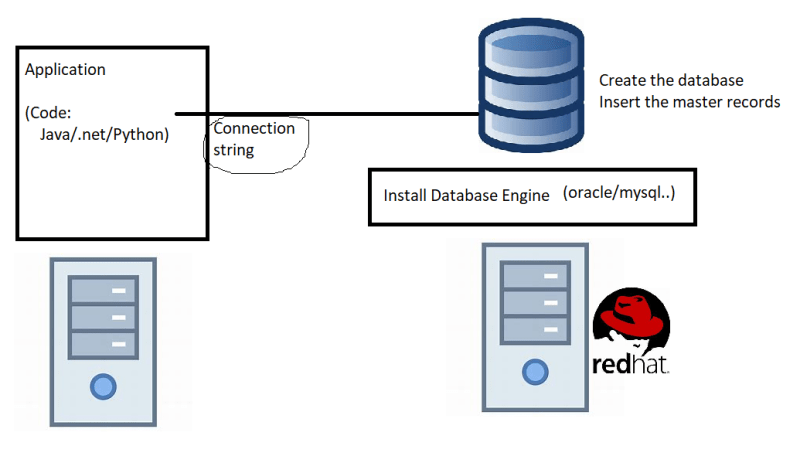
**Databases in an Enterprise**

* Database is generally used to store the master and transactional records of an application.
* Data present in the databases is referred as structure data as we can query it.
* Types of Databases
  + Relational Databases:
    - The data is stored in the tabular format. Each row represents a record
    - To insert, update, delete, query(select) the data we have a language which is referred as SQL (Structured Query Language)
    - The structure of the table often referred as schema is fixed.
    - To use Relational Database we have Database Engines such as
      * Oracle
      * Microsoft SQL Server
      * mySql
      * Postgres
      * IBM DB2
    - Traditional Setup in an enterprise



* + - Steps required are
      * Create a server (physical/virtual) with some os
      * install the DB engine with some version
      * Create a Database
      * Connect from application
    - Maintenance: Generally done by DBA
      * Backup of the data
      * Replicate the data to another db server (for high availability)
      * Tuning the database queries for performance
  + NoSQL Databases:
    - This is a broad category that includes a database that doesn’t use SQL as its primary access language.
    - NoSQL database doesn’t have to confirm to pre-defined schema.
    - This is popular for organizations dealing with semi-structured or unstructured data.
    - Examples:
      * Mongo DB
      * Apache Cassandra
    - The database types that can be derived from NoSQL are
      * Key-Value
      * Document Database
      * Graph Database.
  + Data WareHouse:
    - These type of databases generally deal with large data stores and OLAP/OLTP
    - Examples:
      * Terradata
      * Informatica
* All the databases discussed above store the data on the disk. Disk is bot a very high speed device, There might be some information of your application which doesnot change frequently. RAM is a faster device than disk, If we can store the data in the RAM and when application queries if this infrequently changed data is returned from RAM the application performance increases. Databases which can do this are called Cache-Databases. Examples
  + Redis
  + Memcached.

**Database as a Service**

* Generally when we use Database as a Service (i.e. cloud databases), CSP like AWS
  + will support only few database engines not all.
  + Will support only fewer versions (generally latest)
  + Will give users the option to directly create database (no need to install)
  + Will give options for easier backups and restores (one-click)
  + Will give easier options for replications across datacenters (one-click)
* Charges will be applied based on the size of the engine, DB Engine used. (Hourly charges)
* AWS offers Database as a Service for many different database types  
  